# Circuits over Finite Algebras

Paweł M. Idziak

in collaboration with Piotr Kawałek, Jacek Krzaczkowski and Armin Weiß

THEORETICAL COMPUTER SCIENCE
Jagiellonian University

http://tcs.uj.edu.pl

Panglobal Algebra and Logic Seminar
Boulder & Anywhere on Earth, March 21, 2023

# Solving equations

- Linear equations
- Galois theory
- Diophantine equations – Hilbert's $10^{\text{th}}$ problem
  – studies of decidable subcases
- SAT – satisfiability of Boolean formulas
- . . . and many, many others . . .

- POLSAT – equations of polynomials over (finite) algebras
- SYSPOLSAT – finite systems of polynomial equations over (finite) algebras

# Research project

Brute force algorithm for an equation over a finite algebra $\mathbf{A}$:

$$\mathbf{g}_1(x_1, \ldots, x_n) = \mathbf{g}_2(x_1, \ldots, x_n)$$

requires $|A|^n$ evaluations

**Problem**

*Characterize finite algebras* $\mathbf{A} = (A; f_1, \ldots, f_s)$,
*for which* $\textsc{PolSat}(\mathbf{A})$ *can be solved in polynomial time.*

In which terms such classification is possible?

# Constraint Satisfaction Problem

**CSP over a relational structure $\mathbb{D}$**

asks whether a pp-formula is satisfiable in the structure $\mathbb{D}$

- undecidable in general (e.g. 10th Hilbert problem)
- in NP for finite structures $\mathbb{D}$
- in P or NP-complete for 2-element structures $\mathbb{D}$
  (T.Schaefer, STOC 1978)

**Bulatov (FOCS'17), Zhuk (FOCS'17)**

Constraint satisfaction problem for a fixed finite relational structure is either in P or NP-complete.

# Equations satisfiability and Constraint Satisfaction Problem

## Why bother with equations

– dichotomy for *CSP* is confirmed
– even the precise borderline/characterization is known
– translate it here !!!

## Feder, Madelaine & Stewart 2004; Larose & Zádori 2006

- for every finite relational structure $\mathbb{D}$ there is a finite algebra $\mathbf{A}[\mathbb{D}]$ with $\mathrm{CSP}(\mathbb{D})$ polynomially equivalent to $\textsc{SysPolSat}(\mathbf{A}[\mathbb{D}])$;
- for every finite algebra $\mathbf{A}$ there is a relational structure $\mathbb{D}[\mathbf{A}]$ with $\textsc{SysPolSat}(\mathbf{A})$ polynomially equivalent to $\mathrm{CSP}(\mathbb{D}[\mathbf{A}])$.

## single equation: only one way

- for every finite relational structure $\mathbb{D}$ there is a finite algebra $\mathbf{A}[\mathbb{D}]$ with $\mathrm{CSP}(\mathbb{D})$ polynomially equivalent to $\textsc{PolSat}(\mathbf{A}[\mathbb{D}])$.
- the converse probably not true, unless certain complexity hypothesis fail

# Examples

### Groups (M.Goldmann & A.Russell 1999)

Polynomial satisfiability problem (PolSat)

- is NP-complete for non-solvable groups,
- and in P for nilpotent groups.

### Rings (S.Burris & J.Lawrence 1993; G.Horváth 2011)

For a finite ring **A**, PolSat(**A**) is

- in P, whenever **A** is nilpotent,
- and NP-complete otherwise.

### Lattices (B.Schwarz 2004)

For a finite lattice **A**, PolSat(**A**) is

- in P if **A** is distributive,
- and NP-complete otherwise.

# PolSat is language sensitive

## Case study: non-nilpotent solvable groups

**Fact (Goldmann, Russell)**

PolSat is NP-complete for non-solvable groups
and in P for nilpotent groups.

# PolSat is language sensitive

## Case study: non-nilpotent solvable groups

**Fact (Goldmann, Russell)**

PolSat is NP-complete for non-solvable groups
and in P for nilpotent groups.

**Kosicka Bela observations 2003**

For (solvable but non-nilpotent) symmetric group $S_3$:

- PolSat$(S_3; \cdot, ^{-1})$ is in P (Horváth & Szabó)
- PolSat$(S_3; \cdot, ^{-1}, \text{a couple of additional polynomials})$ is NP-complete.

# PolSat is language sensitive

## Case study: non-nilpotent solvable groups

### Fact (Goldmann, Russell)

PolSat is NP-complete for non-solvable groups
and in P for nilpotent groups.

### Kosicka Bela observations 2003

For (solvable but non-nilpotent) symmetric group $S_3$:

- PolSat$(\mathrm{S}_3; \cdot, {}^{-1})$ is in P (Horváth & Szabó)
- PolSat$(\mathrm{S}_3; \cdot, {}^{-1}$, a couple of additional polynomials$)$ is NP-complete.

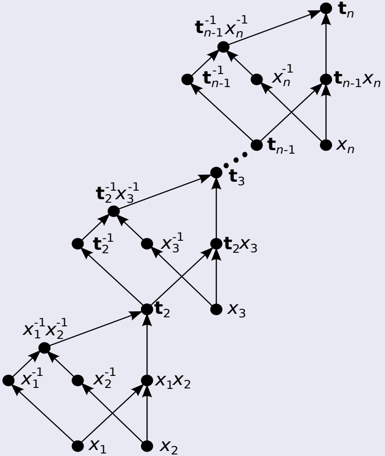
### Fact (Horváth & Szabó 2012)

For (solvable but non-nilpotent) alternating group $A_4$:

- PolSat$(\mathrm{A}_4; \cdot, {}^{-1})$ is in P,
- PolSat$(\mathrm{A}_4; \cdot, {}^{-1}, [,])$, where $[x, y] = x^{-1}y^{-1}xy$,
  is NP-complete.

# exponential syntactic tree vs polynomial size circuit

$$t_n(x_1, x_2, \dots, x_n) = [\dots[[x_1, x_2], x_3] \dots x_n]$$



$\textsc{Csat}(\mathbf{A})$
given a circuit over $\mathbf{A}$ with two output gates $\mathbf{g}_1, \mathbf{g}_2$
is there a valuation of input gates $\overline{x} = (x_1, \ldots, x_n)$ that gives the same output on $\mathbf{g}_1, \mathbf{g}_2$, i.e., $\mathbf{g}_1(\overline{x}) = \mathbf{g}_2(\overline{x})$.

$\textsc{SCsat}(\mathbf{A})$
given a circuit over $\mathbf{A}$ with output gates $\mathbf{g}_1^1, \mathbf{g}_2^1, \ldots, \mathbf{g}_1^k, \mathbf{g}_2^k$
is there a valuation of input gates $\overline{x}$ that gives the same output on all pairs $\mathbf{g}_1^i, \mathbf{g}_2^i$, i.e., $\mathbf{g}_1^i(\overline{x}) = \mathbf{g}_2^i(\overline{x})$ for all $i$.

$\textsc{MCsat}(\mathbf{A})$
given a circuit over $\mathbf{A}$ with output gates $\mathbf{g}_1, \mathbf{g}_2, \ldots, \mathbf{g}_k$
is there a valuation of input gates $\overline{x}$ that gives the same output on all the $\mathbf{g}_i$'s, i.e., $\mathbf{g}_1(\overline{x}) = \mathbf{g}_2(\overline{x}) = \ldots = \mathbf{g}_k(\overline{x})$.

---

$\textsc{Ceqv}(\mathbf{A})$
given a circuit over $\mathbf{A}$ is it true that for all inputs $\overline{x}$ we have the same values on given two output gates $\mathbf{g}_1, \mathbf{g}_2$,
i.e. $\mathbf{g}_1(\overline{x}) = \mathbf{g}_2(\overline{x})$.

# Back to groups

**PolSat (Goldmann & Russell 1999)**

Polynomial satisfiability problem (PolSat)

- is NP-complete for non-solvable groups
- and in P for nilpotent groups.

**CSat (Horváth & Szabó 2011)**

Circuit satisfiability problem (CSat)

- is NP-complete for non-nilpotent groups
- and in P for nilpotent groups.

**Open** (but with some progress)

Characterization of finite groups with poly-time PolSat
in original language, i.e. with multiplication only.

# CSAT in algebras from congruence modular varieties

**LICS'18: two reasons for tractability of CSAT in CM varieties:**

- supernilpotency, (same as nilpotency in groups and rings)
- distributive lattice like behavior.

**LICS'18: many reasons for intractability**

CSAT for algebras not expressible as a product of a nilpotent algebra and a distributive lattice like algebra is NP-complete.

# 1st summary
## LICS'18, SICOMP'22 and STACS'22

| | tractable | open | intractable |
|---|---|---|---|
| CEQV | supernilpotent<br>Aichinger & Mudrinski | nil but not superniI | non nilpotent |
| CSAT | supernil × DL-like | nil but not supernil | non (nil × DL-like) |
| MCSAT | affine × DL-like | — | otherwise |
| SCSAT | affine<br>Gaussian elimination | — | otherwise<br>Larose & Zádori |

Gap for nilpotent but not supernilpotent algebras.

# Hardness part

Easy, moderate and sometimes quite heavy use of:

- comutator theory
- tame congruence theory

A dozen of constructions eliminating bad local behaviours:

- eliminating type **3**
- separating types **2** and **4** (transfer principles)
- forcing type **2** (i.e., solvable) algebras to be nilpotent
- forcing type **4** (subdirectly irreducible) algebras to have only 2 elements

# Poly-time algorithms

For a supernilpotent algebra (or a distributive lattice) **A**
there is a constant $d_{\mathbf{A}}$ so that for each $n$ there is $S_n \subseteq A^n$ with

- $|S_n|$ is $O(n^{d_{\mathbf{A}}})$,
- for two $n$-ary polynomials **s** and **t** the equation $\mathbf{s}(\overline{x}) = \mathbf{t}(\overline{x})$ has a solution $\overline{x} \in A^n$ iff it has a solution in $S_n$.

- in 2-element lattice case: $S_n = \{(0,\ldots,0),(1,\ldots,1)\}$
  in DL-like algebra: $S_n = \{(a,\ldots,a) : a \in A\}$ $\quad d_{\mathbf{A}} = |A|$
- in supernilpotent case:
  $S_n = \bigcup_{a\in A} \{(a_1,\ldots,a_n) : \#\{i : a_i \neq a\} \leqslant d'_{\mathbf{A}}\}, \quad d_{\mathbf{A}} = |A|^{d'_{\mathbf{A}}} \cdot \binom{n}{d'_{\mathbf{A}}}$
  - where (rather huge) constant $d'_{\mathbf{A}}$ is obtained by a quite involved Ramsey type argument,
  - $d'_{\mathbf{A}}$ depends on: size of **A**, supernilpotency degree, functions arity...

After a fascinating race for decreasing $d_{\mathbf{A}}$ we end up with $d_{\mathbf{A}} = 1$,
but for a randomized algorithm

# Nilpotent vs supernilpotent gap

- supernilpotency is not necessary for tractability
- limits of small search space method
- nilpotency is not sufficient for tractability

**(MFCS'18)**

There are nilpotent (but not supernilpotent) algebras **A** with:

- $\textsc{Csat}(\mathbf{A})$ in P,
- $\textsc{Csat}(\mathbf{A})$ can not be solved in polynomial time using algorithm checking a small set of potential solutions which depends only on the number of input gates (unless P = NP).

Example: $\mathbf{A} = (\mathbb{Z}_6;\ +,\ \%2)$

**(LICS'20)**

There are nilpotent algebras **A** with $\textsc{Csat}(\mathbf{A}) \notin$ P, unless ETH fails

**ETH – Exponential Time Hypothesis**

$k$-CNF-SAT requires at least $2^{\sigma_k \cdot n}$ time to be solved
for some constant $0 < \sigma_k \leqslant 1$

## Single equation versus system of equations

- external conjunction in systems of equations
- need to squeeze many terms into a single one
- analogue of an internal conjunction is needed
  - present in Boolean algebras
  - in some rings: $\bigwedge_i t_i = s_i$ iff $\sum_i (t_i - s_i)(t_i - s_i) = 0$
  - solvable non-nilpotent algebras
    have internal (conjunction-like) polynomials of arbitrary arity
    e.g. in groups $[\ldots[[[a, x_1], x_2], x_3]\ldots x_n]$

- Each supernilpotent algebra has its own bound
  for the arity of *conjunction-like* polynomials.
- Nilpotent but not supernilpotent algebras
  do have *conjunction-like* polynomials of arbitrary large arity,
- unfortunately the ones we can construct are of superpolynomial,
  or even exponential size (wrt to the arity).

## Supernilpotent rank of **A**

- splitting congruence lattice into supernilpotent intervals
- supernilpotent algebras have just one such supernilpotent block
- $\mathrm{sr}\,(\mathbf{A}) \leqslant h$ if there is a chain of congruences
  $0_{\mathbf{A}} = \sigma_0 < \sigma_1 < \cdots < \sigma_h = 1_{\mathbf{A}}$
  with $\sigma_{i+1}$ being supernilpotent over $\sigma_i$

## Supernilpotent rank and alternation of primes

For a finite nilpotent algebra **A** from a CM variety tfae:

- $\mathrm{sr}\,(\mathbf{A}) \leqslant h$,
- chains $\varphi_1 < \varphi_2 < \ldots < \varphi_s$ of meet irreducible congruences
  with alternating characteristics (i.e. $\mathrm{char}(\varphi_i) \neq \mathrm{char}(\varphi_{i+1})$),
  have length $s$ bounded by $h$.

- One alternation of primes provides $n$-ary *conjunction-like* polynomials but with exponential length $\Theta(2^{cn})$.

In fact: $(\mathbb{Z}_6;\ +,\ \%2)$ has such polynomials of exactly exponential size

- $h$ alternating primes $p_1 \neq p_2 \neq p_3 \neq \ldots \neq p_h$
  gives $n$-ary *conjunction like* polynomials of length $\Theta(2^{cn^{1/(h-1)}})$,
- which for $h \geqslant 3$ yields subexponential size
- more alternations $\longrightarrow$ shorter conjunction.

**(LICS'20 – examples and an idea of the proof)**

If $\mathbf{A}$ is a finite nilpotent algebra with $\mathrm{sr}(\mathbf{A}) \geqslant 3$ then $\textsc{Csat}(\mathbf{A}) \notin \mathrm{P} \not\ni \textsc{Ceqv}(\mathbf{A})$, actually there are no algorithm for $\textsc{Csat}(\mathbf{A})$ or $\textsc{Ceqv}(\mathbf{A})$ faster than $\Omega\left(2^{c \cdot \log^h n}\right)$

(the first part has been shown in generality by M.Kompatscher, with a very cute proof)

## $CC[p_1; \ldots; p_h]$ modular boolean circuits

$CC[m]$-circuits of depth $h$ with $MOD_{p_i}$ on the $i$-th level
$\mathrm{MOD}_{p_1} \circ \ldots \circ \mathrm{MOD}_{p_h}$

## SESH - Strong Exponential Size Hypothesis (or AND-weakness hypothesis)

The sizes of $CC[p_1; \ldots; p_h]$-circuits, with $h > 1$,
that compute $(\mathrm{AND}_n)_n$, grow at least as $\Omega(2^{cn^{1/(h-1)}})$.

## Deterministic and probabilistic upper bounds (under SESH)

Let **A** be a finite nilpotent algebra from a CM variety with $\mathrm{sr}(\mathbf{A}) = h$.
Then for both $\textsc{CSat}(\mathbf{A})$ and $\textsc{CEqv}(\mathbf{A})$ we have:

- a deterministic $O(2^{c \log^h \ell})$-time algorithm,
  (where $\ell$ is the size of a circuit on the input),
- a probabilistic $O(2^{c \log^{h-1} \ell})$-time algorithm,
  (where $\ell$ is the size of a circuit on the input).

# Further conclusions. . .

## Under ETH & SESH

- no dichotomy for CSAT – in contrast to CSP
- no equivalence of CSAT with CSP – in contrast to SCSAT $\equiv$ CSP
- in fact:
  CSAT has strictly bigger expression power than CSP or SCSAT

## Natural conjecture (at least under ETH)

For a finite algebra **A** from a CM variety
CSAT(**A**) $\in$ P iff **A** is nilpotent and sr (**A**) $\leqslant 2$

Fails. . . but very recently we got:

For a finite algebra **A** from a CM variety
CEQV(**A**) $\in$ RP iff **A** is nilpotent and sr (**A**) $\leqslant 2$

# Barrington, Beigel and Rudich construction

**BBR construction of $CC[m]$-circuits computing $(\mathrm{AND}_n)_n$**

– of depth $\mathbf{3}$,
– and size $2^{O(n^{1/\omega(m)}\cdot\log n)}$,

where $\omega(m)$ is the number of prime divisors of $m$.

**Our recent (LICS'22) improvement of $CC[m]$-circuits computing $(\mathrm{AND}_n)_n$**

– of depth $\mathbf{2}$,
– and size $2^{O(n^{1/\omega(m)}\cdot\log n)}$.

Moreover for any depth $h \geqslant 3$ we have $CC[m]$-circuits computing $(\mathrm{AND}_n)_n$
– of size $2^{O(n^{1/(\omega-1)(h-2)+\omega'}\cdot\log n)}$,

where $\omega'$ is the number of prime divisors of $m$ bigger than $\omega$.

**Consequences for Boolean modular circuits (LICS'22)**

A $CC[m]$-circuit of depth $h$ is satisfiable iff $h = 1$ or $\omega(m) = 1$

# Small supernilpotent rank is not sufficient

CSAT for the algebra $(\mathbb{Z}_{30};\ +;\ \%2)$ is not in P (unless ETH fails)

- higher circuits (bigger $h$)
  $\longrightarrow$ shorter *conjunction-like* polynomials
- wider circuits (i.e. more primes on the same level)
  $\longrightarrow$ shorter *conjunction-like* polynomials
- shorter *conjunction-like* polynomials $\longrightarrow$ bigger complexity

# Group case i.e. PolSat in original language with multiplication only

## CSat

A finite group **G** has CSat in P iff **G** is nilpotent, (unless P =NP), otherwise CSat(**G**) is NP-complete.

## PolSat

- PolSat for nilpotent groups is in P
- PolSat for non-solvable groups is NP-complete
- no solvable group has been known to have NP-complete PolSat
- few examples of solvable, nonnilpotent groups with PolSat in P:
  - $\mathbf{S}_3$, $\mathbf{A}_4$,...
  - all of them have (super)nilpotent (or Fitting) rank 2

Solvable nonnilpotent groups have AND-like polynomials
– but of exponential size in original language of groups

This allows to use methods modelled after nil- but not supernil- realm for CSat

# Towards filling solvable vs nilpotent gap

**(LICS'20, ICALP'20, TOCS'22)**

If $\textsc{PolSat}(\mathbf{G}) \in \mathsf{P}$ then $\mathsf{nr}(\mathbf{G}) \leqslant 2$, unless ETH fails.

**Dihedral groups (LICS'20, ICALP'22)**

For a dihedral group $\mathbf{D}_m$ (with $2m$ elements) we have:

- if $\omega_o(m) \leqslant 1$ then $\textsc{PolSat}(\mathbf{D}_\mathrm{m}) \in \mathsf{RP}$,
- if $\omega_o(m) \geqslant 2$ then $\textsc{PolSat}(\mathbf{D}_\mathrm{m}) \notin \mathsf{RP}$ (under rETH),
- if $\omega_o(m) \geqslant 2$ then $\textsc{PolSat}(\mathbf{D}_\mathrm{m}) \notin \mathsf{P}$ (under ETH),

where $\omega_o(m)$ is the number of odd prime divisors of $m$.

**(ICALP'22)**

If $\mathbf{G}$ has two normal subgroups with

- coprime sizes
- and the join of their centralizers not covering $G$

then $\textsc{PolSat}(\mathbf{G}) \notin \mathsf{RP}$ (under rETH).

# Restricting values for variables in PolSat

ListPolSat – set of possible solutions assigned to each variable
2-ListPolSat – 2-element set of possible solutions assigned to each variable
ProgramSat – 2-element list of possible solutions assigned to each variable, with some connections between these assignments

PolSat $\leqslant_m$ 2-ListPolSat $\leqslant_m$ ListPolSat
2-ListPolSat $\leqslant_m$ ProgramSat

## NUDFA and ProgramSat

Non-uniform deterministic finite automata (over monoids) recognize languages over $\{0, 1\}$
ProgramSat(M) asks if NUDFA's over **M** recognize a nonempty language

## Goldman & Russell

For finite nilpotent groups ProgramSat $\in$ P.
A finite group with ProgramSat $\in$ P has to be solvable.

# Non-Uniform automata or program over algebra **A**

## $n$-ary boolean program $(\mathbf{p}, n, \iota, S)$ over **A**

- $\mathbf{p}$ is a $k$-ary polynomial/circuit over **A**
- $k$ instructions, one for each argument of $\mathbf{p}$ of the form $\iota(x) = (\mathrm{b}^x, a_0^x, a_1^x)$, where $\mathrm{b}^x$ is one of the boolean variables/inputs $\mathrm{b}_1, \ldots, \mathrm{b}_n$, while $a_0^x, a_1^x \in A$,
- set $S \subseteq A$ of accepting values/states.

## Functions associated with program $(\mathbf{p}, n, \iota, S)$

- inner function $(\mathbf{p})[\iota] : \{0,1\}^n \longrightarrow Y$
  $(b_1, \ldots, b_n) \longmapsto \mathbf{p}(a_{b^{x_1}}^{x_1}, \ldots, a_{b^{x_k}}^{x_k})$,
- final $n$-ary boolean function $(\mathbf{p})[\iota, S] : \{0,1\}^n \longrightarrow \{0,1\}$
  with $(\mathbf{p})[\iota, S]\,(b_1, \ldots, b_n) = 1$ iff $(\mathbf{p})[\iota]\,(b_1, \ldots, b_n) \in S$

# ListPolSat and ProgramSat in finite groups

**CDH – constant degree hypothesis** (Barrington, Straubing, and Thérien: Inform&Comput'1990) (Krause, Pudlák: TCS'1997)

$AND_d \circ MOD_m \circ MOD_p$-circuits require $2^{\Omega(n)}$ size to compute $AND_n$ with constant $d$

**Grolmusz and Tardos, SICOMP'2000**

$MOD_m \circ MOD_p$-circuits require $2^{\Omega(n)}$ size to compute $AND_n$

**Barrington, Straubing & Thérien, 1990**

Under CDH:
$\textsc{ProgramSat}(\mathbf{G}_\mathrm{p} \rtimes \mathbf{N}) \in \mathrm{P}$, whenever $\mathbf{G}_p$ is a $p$-group and $\mathbf{N}$ is nilpotent.

**(ICALP'22)**

Under both ETH and CDH:
for a finite solvable group $\mathbf{G}$ with the smallest co-nilpotent normal subgroup $\mathbf{N}$:

$$\textsc{ProgramSat}(\mathbf{G}) \in \mathrm{RP} \quad \text{iff} \quad \mathbf{N} \text{ is a } p\text{-group} \quad \text{iff} \quad \textsc{ListPolSat}(\mathbf{G}) \in \mathrm{RP}$$

# ProgramCSat in finite algebras from CM varieties

**Under both ETH and CDH:** (very recently)

For a finite algebra **A** from a CM variety $\textsc{ProgramCSat}(\mathbf{A}) \in \mathsf{RP}$ iff

- **A** is nilpotent,
- $\mathsf{sr}\,(\mathbf{A}) \leqslant 2$,
- there is only one (prime) characteristics
  below the smallest co-supernilpotent congruence of **A**

**Under both ETH and CDH:** (very recently)

For a finite algebra **A** from a CM variety $\textsc{CEqv}(\mathbf{A}) \in \mathsf{RP}$ iff

- **A** is nilpotent,
- $\mathsf{sr}\,(\mathbf{A}) \leqslant 2$.

**Under both ETH and CDH:** (very recently)

A finite group **G** has $\textsc{PolEqv}(\mathbf{G})$ in RP iff **G** is solvable and $\mathsf{nr}\,(\mathbf{G}) \leqslant 2$,

# Satisfiability in finite lattices

**For a finite lattice $\mathbf{L}$:**

- $\textsc{Csat}(\mathbf{L}) \in P$ iff $\mathbf{L}$ is distributive,
- $\textsc{PolSat}(\mathbf{L}) \in P$ iff $\mathbf{L}$ is distributive,
- $\textsc{ListPolSat}(\mathbf{L}) \in P$ iff $|L| \leqslant 2$,
- $\textsc{ProgramSat}(\mathbf{L}) \in P$ iff $|L| = 1$,

# Open

**CSAT** (even in congruence modular realm)

Which finite nilpotent algebras of supernilpotent rank 2 have CSAT solvable in (randomized) polynomial time?

**POLSAT for groups**

Which finite solvable groups of nilpotent rank 2 have POLSAT solvable in (randomized) polynomial time?

# Open

**CSAT** (even in congruence modular realm)

Which finite nilpotent algebras of supernilpotent rank 2 have CSAT solvable in (randomized) polynomial time?

**PolSat for groups**

Which finite solvable groups of nilpotent rank 2 have PolSat solvable in (randomized) polynomial time?

Thank you

# Open

**CSAT** (even in congruence modular realm)

Which finite nilpotent algebras of supernilpotent rank 2 have CSAT solvable in (randomized) polynomial time?

**POLSAT for groups**

Which finite solvable groups of nilpotent rank 2 have POLSAT solvable in (randomized) polynomial time?

Thank you

and join us